Europäisches Patentamt | European Patent Office | Office européen des brevets

REC'D 06 JAN 2005
WIPO PCT

Bescheinigung / Certificate / Attestation

Die angehefteten Unterlagen stimmen mit der ursprünglich eingereichten Fassung der auf dem nächsten Blatt bezeichneten europäischen Patentanmeldung überein.

The attached documents are exact copies of the European patent application described on the following page, as originally filed.

Les documents fixés à cette attestation sont conformes à la version initialement déposée de la demande de brevet européen spécifiée à la page suivante.

Patentanmeldung Nr. Patent application No. Demande de brevet n°

03027638.0

Der Präsident des Europäischen Patentamts;
Im Auftrag

For the President of the European Patent Office

Le Président de l'Office européen des brevets
p.o.

R C van Dijk

PRIORITY DOCUMENT
SUBMITTED OR TRANSMITTED IN COMPLIANCE WITH RULE 17.1(a) OR (b)

Anmeldung Nr:
Application no.: 03027638.0
Demande no:

Anmeldetag:
Date of filing: 02.12.03
Date de dépôt:

Anmelder/Applicant(s)/Demandeur(s):

DEUTSCHE THOMSON-BRANDT GMBH
Hermann-Schwer-Strasse 3
78048 Villingen-Schwenningen
ALLEMAGNE

Bezeichnung der Erfindung/Title of the invention/Titre de l'invention:
(Falls die Bezeichnung der Erfindung nicht angegeben ist, siehe Beschreibung.
If no title is shown please refer to the description.
Si aucun titre n'est indiqué se referer à la description.)

Method for coding and decoding impulse responses of audio signals

In Anspruch genommene Prioriät(en) / Priority(ies) claimed /Priorité(s) revendiquée(s)
Staat/Tag/Aktenzeichen/State/Date/File no./Pays/Date/Numéro de dépôt:

Internationale Patentklassifikation/International Patent Classification/ Classification internationale des brevets:

G10L/

Am Anmeldetag benannte Vertragstaaten/Contracting states designated at date of filing/Etats contractants désignées lors du dépôt:

AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HU IE IT LU MC NL
PT RO SE SI SK TR LI

Method for coding and decoding impulse responses of audio signals

The invention relates to a method and to an apparatus for coding and decoding impulse responses of audio signals, especially for describing the presentation of sound sources encoded as audio objects according to the MPEG-4 Audio standard.

Background

MPEG-4 as defined in the MPEG-4 Audio standard ISO/IEC 14496-3 and the MPEG-4 Systems standard 14496-1 facilitates a wide variety of applications by supporting the representation of audio objects. For the combination of the audio objects additional information - the so-called scene description - determines the placement in space and time and is transmitted together with the coded audio objects.

For playback the audio objects are decoded separately and composed using the scene description in order to prepare a single soundtrack, which is then played to the listener.

For efficiency, the MPEG-4 Systems standard ISO/IEC 14496-1 defines a way to encode the scene description in a binary representation, the so-called Binary Format for Scene Description (BIFS). Correspondingly, audio scenes are described using so-called AudioBIFS.

A scene description is structured hierarchically and can be represented as a graph, wherein leaf-nodes of the graph form the separate objects and the other nodes describes the processing, e.g. positioning, scaling, effects etc.. The appearance and behavior of the separate objects can be controlled using parameters within the scene description nodes.

Invention

The invention is based on the recognition of the following facts.

The transmission and use of real, i. e. of measured, room impulse responses for the reproduction of sound signals with this room characteristic has been the object of research and development workings for years. Here exist problems with respect to the measurement, post-processing, transmission and the subsequent presentation in the desired listening room.

In the context of the MPEG-4 activities the possibilities of transmission and use of the impulse responses were examined in the frame of the European research project "Carruso". Here, the transmission of long impulse responses turned out to be the main problem. There exist three basic problems:

The necessary length of the pulse responses leads to the problem, that they cannot directly be transmitted as parameter with the field-update-mechanism of MPEG-4.
The transmission with usual coding methods like AAC or MPEG-2/Layer 3 (mp3) results - because of the use of psycho acoustic compression methods - in falsification of the pulse response.
An update-mechanism, which uses the method according to 2., requires sending the pulse responses in the broadcast mode. The bandwidth necessary for this exceeds the bandwidth available for the transport of the media data. Therefore, an update of the data in time cannot be guaranteed.

The present invention describes a mechanism, with which the above-mentioned problems can be solved:

The invention uses multiple successive field updates for the params[128]-field, in order to make complex system parameter

(e. g. system-pulse response) usable in one effect. A first params[128]-field contains information about number and content of the following fields. This represents an extension of the field updates, which is - by default - performed with only one params[128]-field. The transmission of data of any length is made possible. These data can then be stored in an additional memory of a node and they can be used during the calculation of the effect. In principle, it is also possible to replace or amend, respectively, only certain parts of the field during operation, in order to keep the number of transmitted data a small as possible.

<u>Exemplary embodiment</u>

AudioFXProto

Node interface

AudioFxProto {

PROTO audio"Name"[

exposedField	MFNode	**audioFXChildren**	[]	
exposedField	MFFloat	**audioFXParams**	[]	
exposedField	SFInt32	**audioFXnumChannel**	1	
exposedField	MFInt32	**audioFXPhaseGroup**	[]	

]

DEF "Name" AudioFX {

eventIn	MFNode	**addChildren**		
eventIn	MFNode	**removeChildren**		
exposedField	MFNode	**children**	[]	children IS audioFXChildren
exposedField	SFCommand Buffer	**orch**	[]	only used in players with S.A. cap
exposedField	SFCommand Buffer	**score**	[]	only used in players with S.A. cap
exposedField	MFFloat	**params**	[]	params[128] IS audioFXParams
field	SFInt32	**numChannel**	1	numChannel IS audioFXNumCh
field	MFInt32	**phaseGroup**	[]	phaseGroup IS audioFXPhaseG

}

}

Functionality and semantics

The **AudioFxPROTO** node provides an alternative to the **AudioFX** node. It is taylored to consumer products and allows players without Structured Audio capability to use basic audio effects. The PROTO shall encapsulate the **AudioFX** node, so that enhanced MPEG 4 players with Structured Audio capability can decode the SAOL (Structured Audio Orchestra Language - format for the description of instruments) resp. SASL (Structured Audio Score Language - format for the description of scores) token streams directly. Simpler consumer players shall only identify the effects and start them from internal effect representations, if available.

The description of the fields can be found in the description of the **AudioFX** node in the MPEG-4 Systems standard ISO/IEC 14496-1, 9.4.2.10. In short, the **addChildren** eventIn specifies nodes that shall be added to the **children** field, while the **removeChildren** eventIn specifies nodes that shall be removed from the **children** field. The **children** array contains the nodes operated upon by this effect. The **orch** string contains a tokenised block of signal-processing code written in SAOL. The **score** string may contain a tokenized score for the given orchestra written in SASL. The **params** field allows BIFS commands and events to affect the sound-generation process in the orchestra. The **numchan** field specifies the number of channels of audio output by this node. The **phaseGroup** field specifies the phase relationships among the various output channels.

The **BIFS** encoder does not encode the protoName string directly - except if MPEG-J is used, USENAMES keyword - but replaces it by a **PROTO-ID**. This ID will normally be generated automatically during the encoding of the BIFS-stream.

The identification of fixed effects inside a consumer playe: requires reserved IDs. These IDs should - for practical rea· sons - be attached to the corresponding protoName strings i: a special namespace.

The number of possible PROTO-IDs is encoded in the 5-bit variable PROTOIDbits in the BIFSv2Config class. This enables the player to use a maximum of 2^{32} IDs. Reserved IDs shall be located in the "IDs reserved for ISO use" space. For the corresponding protoName strings the JAVA naming convention shall be used.

For reasons of expandability, further ID space should be reserved in a space "IDs reserved for private use", defined by levels. The preserved space could be used by the industry to define their own proprietary nodes.

The protoName strings shall be replaced by their fixed PROTO-IDs in the BIFS encoding process. In case of decoding with a consumer MPEG 4 player, the occurence of these IDs shall cause the BIFS decoder to instantiate the corresponding PROTOs from the matching internal PROTO Effect Nodes.

For a limited number of audio effects, called **Standard Effects**, a Structured Audio code shall to be defined that works directly in enhanced players and is a reference for player-internal implementations. This may be an argument to keep the number of Standard Effects low.

The following **Standard Effects** shall be defined as FX nodes and encapsulated in PROTOs :

Echo, Filter, Stereo-Base, Virtual Stereo, Equalizer, Compressor, Reverb, naturalReverb, Chorus, Flange and Speed-Change (requires media control or (Advanced)AudioBuffer).

PROTO audioNaturalReverb

The audioNaturalReverb contains the following parameters:

First params[ ] field:

	float	*numParamsFields*	1	1..60000
	float	*numImpResp*	0	0..32
	float	*sampleRate*		
	float[ ]	*reverbChannels*	0	0,1,2,3,...,31
	float	*impulseResponseCoding*	0	0..1
	.....			reserved

Following params[ ] fields:

	float	*impulseResponseLength*	0	240000 *
	float[ ]	*impulseResponse*		*
	....			* numImpResp times

The NaturalReverb PROTO uses the impulse responses of different sound channels to create a reverberation effect. Since these impulse responses can be very long (several seconds for a big church or hall), one params[ ] array is not sufficient to transmit the complete data set. Therefore, a bulk of consecutive params[ ] arrays is used in the following way:

The first block of params[ ] contains information about the following params[ ] fields:

The numParamsFields field determines the number of following params[ ] fields to be used. The NaturalReverb PROTO has to provide sufficient memory to store these fields.

The numImpResp defines the number of impulse responses (= number of channels used for reverberation). It must be smaller than **audioFXnumChannel** in the AudioFX PROTO node in-

terface.

The ***reverbChannels*** field defines the mapping of the impuls responses to the input channels.

The ***impulseResponseCoding*** field shows how the impulse response is coded (see table below).

Coding value	Coding function
0	consecutive samples
1	sample-number/sample

Case 1 can be useful to reduce the length of sparse impulse responses.

The fields shall map to the first params[ ] array as follows:

numParamsFields	= params	[0]
numRevChan	= params	[1]
sampleRate	= params	[2]
reverbChannels [0... ***numRevChan*** -1]	= params	[3...3 + ***numRevChan*** - 1]
impulseResponseCoding	= params	[3+***numRevChan***]

The following params[ ] fields contain the ***numImpResp*** consecutive impulse responses as follows:

The ***impulseResponseLength*** gives the length of the following ***impulseResponse***.

The ***impulseResponseLength*** and the ***impulseResponse*** are repeated ***numImpResp*** times.

The fields shall map to the following params[ ] arrays as follows:

impulseResponseLength =params[0]

impulseResponse = params [1... 1 + *impulseResponseLength*] ....

The exact method of calculating the reverberation according to the specified parameters is not normative.

The output shall be the reverberated sound signal.

Claims

1. Method for coding impulse responses of audio signals, wherein said impulse responses allow the reproduction of sound signals corresponding to a certain room characteristic, comprising:

   generating an impulse responses of a sound source; and

   inserting parameters representing said generated impulse responses in multiple successive field updates for the params[128]-field, wherein a first params[128]-field contains information about the number and content of the following fields.

2. Method for decoding impulse responses of audio signals, wherein said impulse responses allow the reproduction of sound signals corresponding to a certain room characteristic, comprising:

   separating parameters representing impulse responses from multiple successive field updates for the params[128]-field, wherein a first params[128]-field contains information about the number and content of the following fields;

   storing the separated parameters in an additional memory of a node; and

   using said stored parameters during the calculation of the room characteristic.

3. Apparatus for performing a method according to claim 1 or 2.

Empfangszeit 2.Dez. 14:28

Abstract

In the context of the MPEG-4 activities the problem of transmitting long impulse responses is solved by inserting parameters representing said generated impulse responses in multiple successive field updates for the params[128]-field, wherein a first params[128]-field contains information about the number and content of the following fields.